

BGPツールあれこれ

- ENOG27 Meeting -

2014年6月20日

株式会社グローバルネットコア

金子 康行

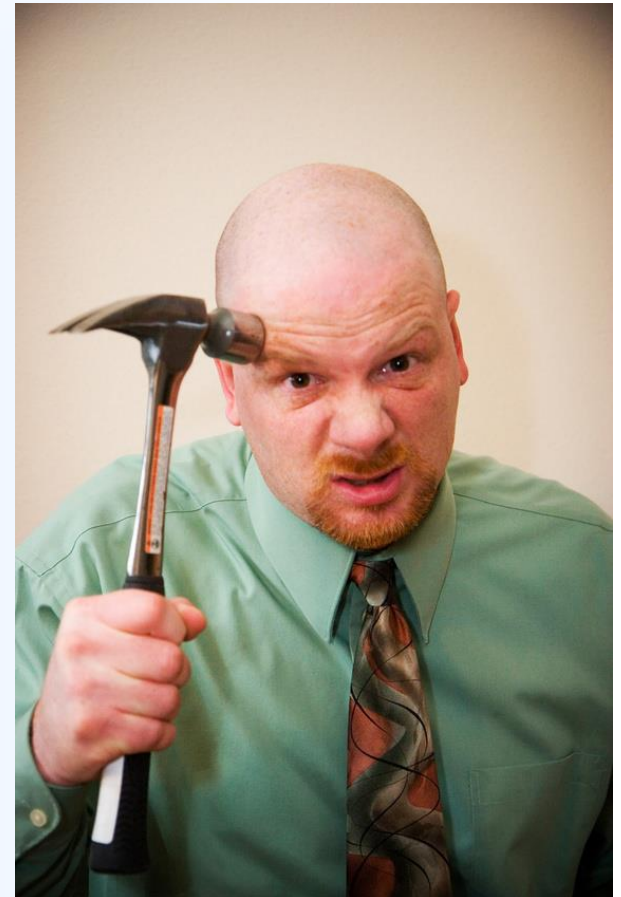
<yasuyuki.kaneko@global-netcore.jp>

✦ みなさん、BGPの検証でどんなツール使ってますか？

✦ 私は……

…… (▮ ▽ ▮) ……

……そもそも検証なんて(ケホケホ



- ✦ 先日。
- ✦ とある事情により。
- ✦ BGPの検証作業が必要になりました。



- ✦ まあ、比較的簡単な検証だったんですけどね……

- ✦ というわけで。
- ✦ せっかくなので。
- ✦ BGP関連のツールについていくつか調べてみました。

- ✦ みなさん既にご存じの内容だったらゴメンナサイ。。。



- ✦ ルータに経路を食わせてゴニョゴニョしたい
- ✦ そんなときはこれを使え！





✦ 概要

- ✦ <http://www.dia.uniroma3.it/~compunet/bgp-probing/>
- ✦ Pythonスクリプトです
 - ✦ Pythonが動く環境なら、たぶんそのまま動きます
- ✦ 対向ルータとpeerを確立し、設定した経路を広告できます
 - ✦ 複雑なことにはできませんが、お手軽に使えます ♪
- ✦ **Announcer was written by Lorenzo Colitti.**
 - ✦ あれ、どこかで聞いたお名前ですね・・・😊

✦ インストール

```
$ wget http://www.dia.uniroma3.it/~compunet/bgp-probing/  
announcer/announcer-1.5.0.tar.gz  
$ tar xvfz announcer-1.5.0.tar.gz  
$ cd announcer-1.5.0  
$ ls  
COPYING  announce.conf.sample  bgppeering.py  peering.py  
README  announce.py            confutil.py  
VERSION  aspathutil.py          iputil.py  
$
```


✦ 設定はこんな感じで

```
$ vi announce.conf
[main]
peerings = Peer1

[DEFAULT]
bgp_id = 192.168.0.1
local_as = 64512
remote_as = 64513
prefixes = 10.0.0.0/16 10.10.0.0/16 10.20.0.0/16 10.30.0.0/16
aspath = 64512

[Peer1]
peer = 192.168.0.2
prefixes = 10.0.0.0/16 10.10.0.0/16 10.20.0.0/16 10.30.0.0/16
```

✦ 起動

```
$ ./announcer.py
```

✦ ログ

```
$ cat announcer.log
[2014/05/23 17:44:39 JST] Starting announcer
[2014/05/23 17:44:39 JST] Peerings configured: Peer1
[2014/05/23 17:44:39 JST] Adding new peering Peer1
[2014/05/23 17:44:39 JST] Peer1: Starting up peering
192.168.0.1 AS64512 -> 192.168.0.2 AS64513
[2014/05/23 17:44:39 JST] Peer1: Sending update for
10.20.0.0/16 with AS path 64512
[2014/05/23 17:44:39 JST] Peer1: Sending update for
10.10.0.0/16 with AS path 64512
[2014/05/23 17:44:39 JST] Peer1: Sending update for
10.30.0.0/16 with AS path 64512
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 10.0.0.0/16	192.168.0.1			0
64512 i				
*> 10.10.0.0/16	192.168.0.1			0
64512 i				
*> 10.20.0.0/16	192.168.0.1			0
64512 i				
*> 10.30.0.0/16	192.168.0.1			0
64512 i				

```
Total number of prefixes 4
```

✦ 設定項目の説明

```
peer:                IPv4 or IPv6 address of the BGP peer.
bgp_id:              The BGP router identifier (an IPv4 address)
local_as:            Local AS number for this peering.
remote_as:           AS number of the peer.
prefixes:            A space-separated list of prefixes to
                     announce to this peer.
                     Note that a prefix will be announced only
                     if it has a non-empty AS-path. Optional.
community:          A space-separated list of BGP community
                     values to announce to this peer
                     (e.g. community = 65501:1234 65123:201).
community[prefix]:  Per-prefix communities. Takes precedence
                     on per-peer and default communities
                     if present.
aspath:              AS-path to announce to this peer (see below).
aspath[prefix]:      Per-prefix AS-path. Takes precedence on
                     per-peer and default AS-path if present.
```

✦ もうちょっと設定を追加してみる

```
[Peer1]
peer = 192.168.0.2
prefixes = 10.0.0.0/16 10.10.0.0/16 10.20.0.0/16 10.30.0.0/16
aspath[10.0.0.0/16] = 64512 64512 64512 ?
aspath[10.10.0.0/16] = 64512 65001 65002 65003 e
aspath[10.20.0.0/16] = 64512 65004 i
community[10.0.0.0/16] = 65535:65281
community[10.10.0.0/16] = 64512:1002
community[10.20.0.0/16] = 64512:1003
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 10.0.0.0/16	192.168.0.1			0
64512 64512 64512 ?				
*> 10.10.0.0/16	192.168.0.1			0
64512 65001 65002 65003 e				
*> 10.20.0.0/16	192.168.0.1			0
64512 65004 i				
*> 10.30.0.0/16	192.168.0.1			0
64512 i				

```
Total number of prefixes 4
```

✦ 相手側のルータで確認

```
$ show ip bgp community no-export
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
* 10.0.0.0/16	192.168.0.1			0
64512 64512 64512 ?				

Total number of prefixes 1

```
$ show ip bgp community 64512:1002
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
* 10.10.0.0/16	192.168.0.1			0
64512 65001 65002 65003 e				

Total number of prefixes 1

✦ 感想

- ✦ とにかくお手軽！
- ✦ 特別なモジュールも不要で、インストールも簡単
- ✦ いくつかの経路を出すだけでいいなら、vyattaなどのソフトウェアルータをわざわざ立てて設定するより楽かも

simple is
beautiful.

✦ 概要

- ✦ <https://code.google.com/p/bgpsimple/>
- ✦ Perlスクリプトです
 - ✦ Net::BGPを利用します
- ✦ 対向ルータとpeerを確立し、経路の送受信をモニタリングできます
 - ✦ BGPDUMPデータを送信経路として食わせることができます
- ✦ (・▽・)イイ!!

✦ インストール

```
$ mkdir bgp_simple; cd bgp_simple
$ wget https://bgpsimple.googlecode.com/files/bgpsimple.tgz
$ tar xvfz bgpsimple.tgz
$ ls
CHANGELOG  README  bgp_simple.pl  bgpsimple.tgz
$
```

✦ 経路データを準備(あとで詳しく説明します)

```
$ wget http://archive.routeviews.org/route-views.wide/bgpdata/
/2014.04/RIBS/rib.20140417.0000.bz2
$ bgpdump -m ./rib.20140417.0000.bz2 > myroutes
$
```

✦ 起動スクリプトをこんな感じで

```
$ vi bgp_simple.sh  
#!/bin/sh
```

```
MYAS=64512
```

```
MYIP=192.168.0.1
```

```
PEERIP=192.168.0.2
```

```
PEERAS=64513
```

```
HOLDTIME=1800
```

```
KEEPALIVE=600
```

```
ROUTEFILE="./myroutes"
```

```
LOGFILE="./bgp_simple.log"
```

```
./bgp_simple.pl -myas $MYAS -myip=$MYIP -peerip=$PEERIP ¥  
-peeras=$PEERAS -holdtime=$HOLDTIME -keepalive=$KEEPALIVE ¥  
-v -p=$ROUTEFILE -o=$LOGFILE.`date +%Y%m%d%H%M`
```

HOLDTIMEを長めに設定しないと
BGPセッションが切れるので注意

✦ 起動

```
$ sudo ./bgp_simple.sh
```

✦ ログ

```
$ tail -f bgp_simple.log.201405231648
Update received from peer [192.168.0.2], ASN [64513]: prfx
[10.0.0.0/16 10.10.0.0/16] aspath [64513]
  nxthp [192.168.0.2] comm [] orig [IGP] agg []
Send Update: prfx [1.0.0.0/24] aspath [64512 2497 15169] orig
[IGP] nxthp [192.168.0.1]
Send Update: prfx [1.0.0.0/24] aspath [64512 7500 2497 15169]
orig [IGP] nxthp [192.168.0.1]
Send Update: prfx [1.0.4.0/24] aspath [64512 7500 2516 6453
7545 56203] orig [IGP] nxthp [192.168.0.1]
Send Update: prfx [1.0.4.0/24] aspath [64512 2497 6453 7545
56203] orig [IGP] nxthp [192.168.0.1]
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 1.0.0.0/24	192.168.0.1			0
64512 7500 2497 15169 i				
*> 1.0.4.0/24	192.168.0.1			0
64512 2497 6453 7545 56203 i				
*> 1.0.5.0/24	192.168.0.1			0
64512 2497 6453 7545 56203 i				
*> 1.0.6.0/24	192.168.0.1			0
64512 7500 2497 4826 38803 56203 i				
*> 1.0.7.0/24	192.168.0.1			0
64512 7500 2497 4826 38803 56203 i				
*> 1.0.20.0/23	192.168.0.1			0
64512 7500 2519 i				

✦ 使い方の説明

```
usage:
bgp_simple.pl:

    -myas    ASNUMBER
             # (mandatory) our AS number
    -myip    IP address
             # (mandatory) our IP address to source the
             session from
    -peerip  IP address
             # (mandatory) peer IP address
    -peeras  ASNUMBER
             # (mandatory) peer AS number
```

✦ 使い方の説明

```
[-holdtime]      Seconds
                  # (optional) BGP hold time duration in seconds
                  (default 60s)
[-keepalive]     Seconds
                  # (optional) BGP KeepAlive timer duration
                  in seconds (default 20s)

[-v]
                  # (optional) provide verbose output to STDOUT,
                  use twice to get debugs

[-p file]
                  # (optional) prefixes to advertise
                  (bgpdump formatted)

[-o file]
                  # (optional) write all sent and received UPDATE
                  messages to file
```


✦ 使い方の説明

```
[-m number]
    # (optional) maximum number of prefixes to
    advertise
[-n IP address]
    # (optional) next hop self, overrides original
    value
[-l number]
    # (optional) set default value for LOCAL_PREF
[-dry]
    # (optional) dry run; dont build adjacency,
    but check prefix file (requires -p)
```

✦ 使い方の説明

```
[-f KEY=REGEX]
# (optional) filter on input prefixes (requires -p),
# repeat for multiple filters
KEY is one of the following attributes
(CaSe insensitive):
```

NEIG	originating neighbor
NLRI	NLRI/prefix(es)
ASPT	AS_PATH
ORIG	ORIGIN
NXHP	NEXT_HOP
LOCP	LOCAL_PREF
MED	MULTI_EXIT_DISC
COMM	COMMUNITY
ATOM	ATOMIC_AGGREGATE
AGG	AGGREGATOR

```
REGEX is a perl regular expression to be expected
in a match statement (m/REGEX/)/)
```

✦ たとえばこんなことも

```
$ sudo ./bgp_simple.pl -myas 64512 -myip=192.168.0.1 -
peerip=192.168.0.2 -peeras=64513 -holdtime=1800 -keepalive=600
-p myroutes -v -f ORIG=INCOMPLETE
----- CONFIG SUMMARY -----
Configured for an eBGP session between me (ASN64512,
192.168.0.1) and peer (ASN64513, 192.168.0.2).
Using 600 seconds as KeepAlive value and 1800 seconds as
HoldTime value for this peer.
Generating verbose output, level 1.
Will use prefixes from file myroutes.
Will set next hop address to 192.168.0.1 because of eBGP
peering.
Will apply filter to input file:
      ORIG =~ /INCOMPLETE/
-----
Connection established with peer 192.168.0.2, AS 64513.
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 1.9.52.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				
*> 1.9.53.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				
*> 1.9.54.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				
*> 1.9.55.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				
*> 1.9.112.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				
*> 1.9.113.0/24	192.168.0.1			0
64577 7500 2516 4788 ?				

✦ 感想

- ✦ 名前のとおり、とてもシンプル！
- ✦ インストールも動かすのもさして難しくない
- ✦ BGPDUMPのデータを食わせることができるので、リアルなインターネット経路(フルルート)で検証することができるのがいい
- ✦ 受信経路の変化もリアルタイムにモニタできるよ
- ✦ というわけで、おススメです☆



✦ 概要

- ✦ <http://elxsi.de/inject/doku.php>
- ✦ Perlスクリプトです
 - ✦ 以下のモジュールを利用します
 - ✦ Net::BGP、Term::ShellUI、Term::ReadLine
 - ✦ XML::Simple、IO::Interface、Data::Dumper
- ✦ 対向ルータとpeerを確立し、対話型インタフェースを使って様々な操作を行ったり、状態を確認したいすることができます

✦ インストール

```
$ wget http://elxsi.de/inject/lib/exe/fetch.php/inject-0.01.tar.gz
$ tar xvfz inject-0.01.tar.gz
$ cd inject
$ ls
Inject  cfg  inject.pl  patch  tests
$ sudo cp inject.pl /usr/local/bin/
$ sudo chmod +x /usr/local/bin/inject.pl
$ mkdir ~/.inject
$ cp cfg/inject.rc ~/.inject/
```


✦ 設定はこんな感じで

```
$ vi ~/.inject/inject.rc
<local name="Local">
    <address>192.168.0.1</address>
    <as>64512</as>
</local>
<peer name="Peer1">
    <description>Connection to Peer1</description>
    <address>192.168.0.2</address>
    <port>179</port>
    <as>64513</as>
    <holdtime>1800</holdtime>
    <connectretrytime>20</connectretrytime>
    <keepalivetime>600</keepalivetime>
    <activate>1</activate>
    <listen>1</listen>
    <passive>0</passive>
</peer>
```

✦ 起動

```
$ sudo /usr/local/bin/inject.pl ~/.inject/inject.rc
```

```
Inject v0.01 - (c) by Martin Kluge <mk@elxsi.de>
```

```
=====
```

```
Type "help", "h" or "?" for command overview.
```

```
Inject>
```

✦ 対話型インタフェースですよ

```
Inject> h
        debug -- Debugging options
        exit  -- Exit program
        flap  -- Flap peers and routes
generate -- Generate random routes
        help  -- Help
history   -- Prints the command history
inject    -- Inject routes
        peer  -- Start / stop peers
        route -- Set route options
        show  -- Show commands
        test  -- Test commands
unflap    -- Unflap peers and routes
withdraw  -- Withdraw routes

Inject>
```

✦ Peerの状態を確認

```
Inject> show peers
Jun 1 12:01:36: INFO: BGP peer summary:

Local address      : 192.168.0.1
Local AS           : 64512
Number of peers    : 1
Number of updates  : 1
Number of NLRI's   : 2
Number of withdrawns: 0
Recvd prefixes     : 2
Sent prefixes      : 0

PeerID      Neighbor      V  T   AS      State
PfxRecvd    PfxSent
Peer1       192.168.0.2    4  E   64513   Established  2
0
```

Inject>

✦ 使い方

Command	Description
show config	Show config
show debug (<arg>)	Show debugging options
show peer <peerid ip address remote asn>	Show detailed peer information
show peers	Show peer overview
show route <peerid all> <route>	Show detailed route information
show routes <peerid all>	Show route overview
show sentroute <peerid all> <route>	Show detailed information of sent routes
show sentroutes <peerid all>	Show overview information of sent routes

✦ 使い方

Command	Description
inject <peerid all> <rid>	Inject route with specified RID on the specified peer
route aggregator <rid> <asn> <aggregator ip>	Set route aggregator
route aspath <rid> <as1>...<asN>	Set AS path
route atomic <rid> <0 1>	Set ATOMIC_AGGREGATE
route community <rid> <c1>...<cN>	Set community
route localpref <rid> <localpref>	Set localpref
route med <rid> <med>	Set MED
route net <rid> <network>	Set prefix
route nexthop <rid> <nexthop>	Set next-hop
route origin <rid> <1 2 3>	Set origin (0=IGP, 1=EGP, 2=INCOMPLETE)
route remove <rid all>	Remove route (will be withdrawn if it is currently injected)
route show <rid all>	Show route information

✦ 使い方

Command	Description
withdraw aggregator <peerid all> <asn ip>	Withdraw routes matching aggregator
withdraw all (<peerid>)	Withdraw all routes
withdraw aspath <peerid all> <as1>...<asN>	Withdraw routes matching AS path
withdraw atomic <peerid all> <0 1>	Withdraw routes matching ATOMIC_AGGREGATE
withdraw community <peerid all> <c1>...<cN>	Withdraw routes matching community
withdraw localpref <peerid all> <localpref>	Withdraw routes matching localpref
withdraw med <peerid all> <med>	Withdraw routes matching MED
withdraw nexthop <peerid all> <nexthop>	Withdraw routes matching nexthop
withdraw origin <peerid all> <origin>	Withdraw routes matching origin
withdraw rid <peerid all> <rid>	Withdraw a specific route
withdraw route <peerid all> <route>	Withdraw route matching prefix

✦ 使い方

Command	Description
generate remove	Removes all generated routes from all peers
generate routes <peerid all> <num> (<args>)	Generate and inject a number <num> of routes
flap peer <peerid all> <up_s> <down_s>	Flapps peer, up_s is the number of seconds a peer should stay up, down_s is the number of seconds a peer should stay down
flap route <peerid all> <rid> <up_s> <down_s>	Flapps route, up_s is the number of seconds a route should stay up, down_s is the number of seconds a route should stay down
unflap peer <peerid all>	Stop peer flapping, peer will stay in last flap state
unflap route <peerid all> <rid>	Stop route flapping, route will stay in last flap state

✦ 使い方

Command	Description
peer start <peerid all>	Start peer
peer stop <peerid all>	Stop peer
test start <testfile> <outputfile>	Start test
test sleep <seconds>	Wait some seconds
test waitfor <seconds> “<regexp>”	Wait for match

✦ 経路を広告してみる

```
Inject> route net 1 10.0.0.0/16
Jun 1 12:25:23: INFO: Route network attribute for RID 1 set.

Inject> route aspath 1 64512
Jun 1 12:26:12: INFO: Route AS path attribute for RID 1 set.

Inject> route nexthop 1 192.168.0.1
Jun 1 12:26:50: INFO: Route nexthop attribute for RID 1 set.

Inject> inject Peer1 1
Jun 1 12:27:00: INFO: Injecting the following route:
RID          : 1
Inject to    : Peer1
Network      : 10.0.0.0/16
NextHop      : 192.168.0.1
ASPath       : 64512

Jun 1 12:27:00: Injecting RID 1 on Peer1.
Jun 1 12:27:01: INFO: Injecting RID 1 on Peer1
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 10.0.0.0/16	192.168.0.1	0		0
64512 i				

```
Total number of prefixes 1
```

✦ 経路情報を変更してみる

```
Inject> route aspath 1 64512 64512 64512
Jun 1 12:34:38: INFO: Route AS path attribute for RID 1 set.

Inject> route med 1 200
Jun 1 12:35:02: INFO: Route MED attribute for RID 1 set.

Inject> inject Peer1 1
Jun 1 12:36:13: INFO: Injecting the following route:
RID           : 1
Inject to     : Peer1
Network       : 10.0.0.0/16
NextHop       : 192.168.0.1
ASPath        : 64512 64512 64512
MED           : 200

Jun 1 12:36:13: Route RID 1 already injected on Peer1.
Reinjecting it.
Jun 1 12:36:14: INFO: Withdrawing and reinjecting RID 1 on
Peer1.
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 10.0.0.0/16	192.168.0.1	200		0
64512 64512 64512 i				

```
Total number of prefixes 1
```

✦ generate routesコマンドでランダムな経路を生成可能

```
Usage: generate routes <peerid|all> <number of routes>  
<args1>...<argsN>
```

Valid arguments are:

flap(0-100)	-> Percent of routes which should flap Flap time is between 1 and 120 secs
nexthop(<nh1> ...)	-> Nexthops
origin(0 1 2)	-> Origin
localpref(<l1> ...)	-> LocalPref
med(<med1> ...)	-> Multi-exit discriminator (MED)
atomic(0 1)	-> Atomic aggregate
aggregator(<asn1:agg1> ...)	-> Aggregator
aspath(<as1,as2> ...)	-> AS Path
community(<aa1:dd1,aa2:dd2> ...)	-> Communities

✦ ランダム経路の生成、やってみよう

```
Inject> generate routes Peer1 100 nexthop(192.168.0.1)  
aspath(64512,65001,65002|64512,65004) med(0)
```

```
Jun 1 15:22:28: INFO: Generating 100 routes. One dot for each  
1000 routes.
```

✦ 相手側のルータで確認

```
$ show ip bgp neighbors 192.168.0.1 received-routes
```

Network	Next Hop	Metric	LocPrf	Weight
Path				
*> 0.0.0.0	192.168.0.1	0		0
64512 65001 65002 i				
*> 0.0.0.0	192.168.0.1	0		0
64512 65001 65002 e				
*> 1.252.170.0/17	192.168.0.1	0		0
6451265001 65002 e				
*> 5.195.228.0/17	192.168.0.1	0		0
64512 65004 i				
*> 12.246.181.128/31	192.168.0.1	0		0
64512 65004 i				
*> 16.0.0.0/5	192.168.0.1	0		0
64512 65004 ?				

✦ 相手側のルータで確認

```
*> 17.73.0.0/9      192.168.0.1      0      0
64512 65004 ?
*> 19.0.0.0/7       192.168.0.1      0      0
64512 65004 i
*> 18.43.178.77/26  192.168.0.1      0      0
64512 65004 ?
*> 19.76.191.11/30  192.168.0.1      0      0
64512 65004 ?
*> 24.189.210.0/20  192.168.0.1      0      0
64512 65001 65002 ?
*> 25.17.110.229/30 192.168.0.1      0      0
64512 65001 65002 i
*> 29.0.0.0         192.168.0.1      0      0
64512 65004 e
*> 30.28.3.0/22     192.168.0.1      0      0
64512 65001 65002 e
*> 32.0.0.0/4       192.168.0.1      0      0
64512 65004 e
```

✦ flappingを起こしてみる

```
Inject> show sentroutes all
```

RID	S	Network	NextHop	Peer	PeerID
1	I	10.0.0.0/16	192.168.0.1	192.168.0.2 /	Peer1
3	I	10.20.0.0/16	192.168.0.1	192.168.0.2 /	Peer1
2	I	10.10.0.0/16	192.168.0.1	192.168.0.2 /	Peer1

```
Inject> flap route Peer1 1 5 5
```

```
Jun 1 16:28:12: INFO: Flapping for RID 1 on peer Peer1 enabled  
(UP=5 / DOWN=5)
```

```
Jun 1 16:28:12: INFO: Flapping for RID 3 on peer Peer1 enabled  
(UP=5 / DOWN=5)
```

```
Jun 1 16:28:12: INFO: Flapping for RID 2 on peer Peer1 enabled  
(UP=5 / DOWN=5)
```

RID1に対して設定を入れているにも関わらず、
なぜかRID2とRID3にも同じ設定が適用される。
バグか??

✦ flappingを起こしてみる

```
Jun 1 16:28:18: INFO: Flap time for route 1 is over ->
Withdrawing...
Jun 1 16:28:18: INFO: Flap time for route 3 is over ->
Withdrawing...
Jun 1 16:28:18: INFO: Flap time for route 2 is over ->
Withdrawing...
Jun 1 16:28:18: INFO: Withdrawing RID 1 on Peer1
Jun 1 16:28:18: INFO: Withdrawing RID 3 on Peer1
Jun 1 16:28:18: INFO: Withdrawing RID 2 on Peer1
Jun 1 16:28:24: INFO: Flap time for route 1 is over ->
Starting...
Jun 1 16:28:24: INFO: Flap time for route 3 is over ->
Starting...
Jun 1 16:28:24: INFO: Flap time for route 2 is over ->
Starting...
Jun 1 16:28:24: INFO: Withdrawing and reinjecting RID 1 on
Peer1.
Jun 1 16:28:24: INFO: Withdrawing and reinjecting RID 3 on
Peer1.
```

✦ flappingを起こしてみる

```
Jun 1 16:28:24: INFO: Withdrawing and reinjecting RID 2 on
Peer1.
Jun 1 16:28:30: INFO: Flap time for route 1 is over ->
Withdrawing...
Jun 1 16:28:30: INFO: Flap time for route 3 is over ->
Withdrawing...
Jun 1 16:28:30: INFO: Flap time for route 2 is over ->
Withdrawing...
Jun 1 16:28:30: INFO: Withdrawing RID 1 on Peer1
Jun 1 16:28:30: INFO: Withdrawing RID 3 on Peer1
Jun 1 16:28:30: INFO: Withdrawing RID 2 on Peer1
Jun 1 16:28:36: INFO: Flap time for route 1 is over ->
Starting...
Jun 1 16:28:36: INFO: Flap time for route 3 is over ->
Starting...
Jun 1 16:28:36: INFO: Flap time for route 2 is over ->
Starting...
Jun 1 16:28:36: INFO: Withdrawing and reinjecting RID 1 on
Peer1.
```

✦ 相手側のルータで確認

```
$ show ip bgp dampened-paths
```

Network	From	Reuse	Path
*d 10.0.0.0/16	192.168.0.1	00:49:01	64512 i
*d 10.10.0.0/16	192.168.0.1	00:51:37	64512 65001 65002 i
*d 10.20.0.0/16	192.168.0.1	00:49:04	64512 65004 i

```
Total number of prefixes 3
```

✦ 感想

- ✦ いろんなモジュール使っているのでインストールはやや面倒
- ✦ ちょっと動きが不安定な印象...
- ✦ 対話型インタフェースはやっぱり便利
- ✦ リアルタイムに経路操作ができるのがいい感じ
- ✦ generate機能は便利だけど、現実にはありえないような経路が出てきてしまうので、ちょっと気持ちが悪い
- ✦ test機能を使うと、あらかじめ作ったシナリオ(コマンドリスト)を実行することができるらしい(未検証)
 - ✦ `test start <testfile> <outputfile>`



Peta

Tera

Giga

Mega

Kilo

✦ 概要

- ✦ <https://github.com/Exa-Networks/exabgp>
- ✦ Pythonスクリプトです
 - ✦ Python以外に特に必要なものはありません
 - ✦ ただし、最新版をPython2.7未満で動かすには、`argparse`モジュールの追加が必要？
- ✦ 対向ルータとpeerを確立し、経路の送受信を行うことができます
 - ✦ `configuration`ファイルはJuniperライクな書式

✦ インストール

```
$ wget https://github.com/Exa-Networks/exabgp/archive/  
3.3.2.tar.gz  
$ tar xvfz 3.3.2.tar.gz  
$ cd exabgp-3.3.2  
$ ls  
CHANGELOG  PEP8          ROADMAP  dev  lib  service  
COPYRIGHT  README.md    debian  etc  sbin  setup.py
```

✦ そのままでは動かなかったので追加作業・・・

```
$ cat /etc/redhat-release  
CentOS release 6.5 (Final)  
$ python --version  
Python 2.6.6  
$ sudo easy_install pip  
$ sudo pip install argparse
```

✦ そのままでも起動できるけど

```
$ pwd
/home/yasuyuki/exabgp-3.3.2
$ ./sbin/exabgp --version
3.3.2
```

✦ 「インストール」することもできますよ

```
$ python setup.py build
$ sudo python setup.py install
$ which exabgp
/usr/bin/exabgp
$ ./sbin/exabgp --version
3.3.2
```

✦ 設定はこんな感じで

```
$ vi exabgp.conf
neighbor 192.168.0.2 {
    description "peer1";
    router-id 192.168.0.1;
    local-address 192.168.0.1;
    local-as 64512;
    peer-as 64513;
    hold-time 1800;
    static {
        route 10.0.1.0/24 {
            next-hop 192.168.0.1;
        }
        route 10.0.2.0/24 {
            next-hop 192.168.0.1;
            community :30740;
        }
    }
}
```

✦ 起動

```
$ exabgp exabgp.conf
```

✦ ログ

```
Thu, 19 Jun 2014 10:26:39 | INFO      | 15544 | reactor      |
Performing reload of exabgp 3.3.2
Thu, 19 Jun 2014 10:26:39 | INFO      | 15544 | reactor      |
New Peer neighbor 192.168.0.2 local-ip 192.168.0.1 local-as
64512 peer-as 64513 router-id 192.168.0.1 family-allowed in-
open
Thu, 19 Jun 2014 10:26:39 | WARNING   | 15544 | configuration |
Loaded new configuration successfully
Thu, 19 Jun 2014 10:26:39 | INFO      | 15544 | network      |
Connected to peer neighbor 192.168.0.2 local-ip 192.168.0.1
local-as 64512 peer-as 64513 router-id 192.168.0.1 family-
allowed in-open (out)
```

✦ プロセスにシグナルを送ることで設定を反映

The program configuration can be controlled using signals:

- SIGLARM : restart ExaBGP
- SIGUSR1 : reload the configuration
- SIGUSR2 : reload the configuration and the forked processes
- SIGTERM : terminate ExaBGP
- SIGHUP : terminate ExaBGP (does NOT reload the configuration anymore)

✦ たとえば設定ファイルを編集して再読み込みするなら

```
$ vi exabgp.conf  
$ kill -SIGUSR1 [pid]
```

✦ 環境変数を使って詳細なオプション設定が可能

Individual configuration options can be set using environment variables, such as :

```
> env exabgp.daemon.daemonize=true ./sbin/exabgp  
or > env exabgp.daemon.daemonize=true ./sbin/exabgp  
or > export exabgp.daemon.daemonize=true; ./sbin/exabgp
```

Environment values are:

- exabgp.api.encoder
- exabgp.bgp.openwait
- exabgp.cache.attributes
- exabgp.cache.nexthops
- exabgp.daemon.daemonize
- exabgp.daemon.pid
- exabgp.daemon.user
- exabgp.log.all
- exabgp.log.configuration
- exabgp.log.daemon
- exabgp.log.destination

<snip>

✦ 感想

- ✦ 感想書くほど触ってないですごめんなさい
- ✦ 大量の経路を注入するだけなら、bgpsimpleのほうがお手軽？
 - ✦ BGPDUMPのデータをExaBGPのconfigに変換するツールなどもあるみたいです
 - ✦ <http://aimless.jp/blog/archives/1991>
- ✦ ExaBGPをベースに、必要な機能を追加開発して利用している事業者さんもあるみたい
 - ✦ JANOG33でMicrosoftさんが発表したSDN的な利用事例
 - ✦ <http://www.janog.gr.jp/meeting/janog33/program/bgp.html>

- ✦ 実際に流れている経路を参照して分析したい
- ✦ そんなときはこれを使え！



Route Views



✦ 概要

- ✦ <http://www.routeviews.org/>
- ✦ オレゴン大学が運営するプロジェクト
- ✦ グローバルインターネットの経路情報が異なる地点でどのように見えているか、リアルタイムに把握するための取り組み
 - ✦ telnetまたはSSHで様々なルータの経路情報にアクセスすることが可能
- ✦ 2時間ごとに出力したMRTフォーマットのダンプデータがアーカイブされている
 - ✦ <http://archive.routeviews.org/>

Route Views Services

• Services

Route-Views offers the following services:

Host	MFG	BGP Proto	UI	Location
route-views.routeviews.org (route-views.oregon-ix.net)	Cisco	IPv4 uni/multi-cast multi-hop	telnet	U of Oregon, Eugene Oregon, USA
route-views2.routeviews.org	Quagga	IPv4 uni/multi-cast multi-hop	telnet	U of Oregon, Eugene Oregon, USA
route-views3.routeviews.org	Quagga	IPv4 uni/multi-cast multi-hop	telnet	U of Oregon, Eugene Oregon, USA
route-views4.routeviews.org	Quagga	IPv4/IPv6 uni/multi-cast multi-hop	telnet	U of Oregon, Eugene Oregon, USA
route-views6.routeviews.org	Zebra	IPv6 multi-hop	telnet	U of Oregon, Eugene Oregon, USA
route-views.eqix.routeviews.org	Zebra	IPv4/v6 uni/multi-cast non-multi-hop	telnet	Equinix, Ashburn, VA
route-views.isc.routeviews.org	Zebra	IPv4/v6 uni/multi-cast non-multi-hop	telnet	ISC (PAIX), Palo Alto CA, USA
route-views.kixp.routeviews.org	Zebra	IPv4 uni/multi-cast non-multi-hop	telnet	KIXP, Nairobi, Kenya
route-views.linx.routeviews.org	Zebra	IPv4/v6 uni/multi-cast non-multi-hop	telnet	LINX, London, GB
route-views.nwax.routeviews.org	Zebra	IPv4/v6 uni/multi-cast non-multi-hop	telnet	NWAX, Portland, Oregon
route-views.wide.routeviews.org	Zebra	IPv4/6 uni/multi-cast non-multi-hop	telnet	DIXIE (NSPIX), Tokyo, Japan
route-views.sydney.routeviews.org	Zebra	IPv4/6 uni/multi-cast non-multi-hop	telnet	SYDNEY (SYD1 Equinix), Sydney, Australia
route-views.saopaulo.routeviews.org	Zebra	IPv4/6 uni/multi-cast non-multi-hop	telnet	SAOPAULO (PTT Metro, NIC.br), Sao Paulo, Brazil
route-views.telxatl.routeviews.org	Zebra	IPv4/6 uni/multi-cast non-multi-hop	telnet	TELXATL (TELX Atlanta), Atlanta, Georgia

Access to Route Views

```
$ telnet route-views.wide.routeviews.org
Trying 203.178.141.138...
Connected to route-views.wide.routeviews.org.
Escape character is '^['.
```

```
Hello, this is Quagga (version 0.99.17).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
route-views.wide.routeviews.org> show ip bgp
BGP table version is 0, local router ID is 202.249.2.166
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 1.0.0.0/24	202.249.2.169	0	2497	15169	i
*	202.249.2.86	0	7500	2497	15169 i
* 1.0.4.0/24	202.249.2.86	0	7500	2497	6453 7545 56203 i
*>	202.249.2.169	0	2497	6453	7545 56203 i
* 1.0.5.0/24	202.249.2.86	0	7500	2497	6453 7545 56203 i
*>	202.249.2.169	0	2497	6453	7545 56203 i
*> 1.0.6.0/24	202.249.2.169	0	2497	4826	38803 56203 i
*	202.249.2.86	0	7500	2497	4826 38803 56203 i
*> 1.0.7.0/24	202.249.2.169	0	2497	4826	38803 56203 i
*	202.249.2.86	0	7500	2497	4826 38803 56203 i
* 1.0.20.0/23	202.249.2.169	0	2497	2519	i
*>	202.249.2.86	0	7500	2519	i





















University of Oregon Route Views Archive Project

David Meyer

- Please see www.routeviews.org for a description of the route views project, bibliography, and additional information.
- For asn.routeviews.org zone files [click here](#) or ftp from: [ftp.routeviews.org/dnszones/](ftp://ftp.routeviews.org/dnszones/)
- Data Archives
 - [MRT format RIBs and UPDATES](#) (quagga bgpd, from route-views2.oregon-ix.net)
 - [MRT format RIBs and UPDATES](#) (quagga bgpd, from route-views3 as of Aug 13, 2013)
 - [MRT format RIBs and UPDATES](#) (quagga bgpd, from route-views4.routeviews.org)
 - [v6 MRT format RIBs and UPDATES](#) (quagga bgpd, from route-views6.oregon-ix.net)
 - [MRT format RIBs and UPDATES from Equinix Ashburn](#) (quagga bgpd, from route-views.eqix.routeviews.org)
 - [MRT format RIBs and UPDATES from ISC \(PAIX\)](#) (quagga bgpd, from route-views.isc.routeviews.org)
 - [MRT format RIBs and UPDATES from KIXP](#) (quagga bgpd, from route-views.kixp.routeviews.org)
 - [MRT format RIBs and UPDATES from JINX](#) (quagga bgpd, from route-views.jinx.routeviews.org)
 - [MRT format RIBs and UPDATES from LINX](#) (quagga bgpd, from route-views.linx.routeviews.org)
 - [MRT format RIBs and UPDATES from NWAX](#) (quagga bgpd, from route-views.nwax.routeviews.org)
 - [MRT format RIBs and UPDATES from TELXATL](#) (quagga bgpd, from route-views.telxatl.routeviews.org)
 - [MRT format RIBs and UPDATES from DIXIE \(WIDE\)](#) (quagga bgpd, from route-views.wide.routeviews.org)
 - [MRT format RIBs and UPDATES from SYDNEY](#) (quagga bgpd, from route-views.sydney.routeviews.org)
 - [MRT format RIBs and UPDATES from SAOPAULO](#) (quagga bgpd, from route-views.saopaulo.routeviews.org)
 - [ipv6 data split out from the above files](#) (multiple collectors)
 - ['sh ip bgp' format RIBs](#) from route-views.route-views.org (to now)
 - [route dampening data](#) from route-views.route-views.org (to March 2008)
 - ['sh ip bgp' format RIBs](#) from route-views3.routeviews.org (to May 2012)
 - [route dampening data](#) from route-views3.routeviews.org (to August 2012)
 - The [collector script](#) that gathers the Cisco data was writted by Sean McCreary.

Note: MRT RIB and UPDATE files have internal timestamps in the standard Unix format, however the file names are constructed based on the time zone setting of the collector. The collectors had their time zones set to Pacific Time prior to Feb 3, 2003 at approximately 19:00 UTC. At that time all but one of the existing collectors had their time zones reset to UTC. The one exception was routeviews.eqix which was not reset to UTC until Feb 1, 2006 at approximately 21:00 UTC.

Index of /route-views.wide/bgpdata/2014.04/RIBS

Name	Last modified	Size	Description
 Parent Directory	-		
 rib.20140401.0000.bz2	01-Apr-2014 00:00	5.6M	
 rib.20140401.0200.bz2	01-Apr-2014 02:00	5.6M	
 rib.20140401.0400.bz2	01-Apr-2014 04:00	5.6M	
 rib.20140401.0600.bz2	01-Apr-2014 06:00	5.6M	
 rib.20140401.0800.bz2	01-Apr-2014 08:00	5.6M	
 rib.20140401.1000.bz2	01-Apr-2014 10:00	5.6M	
 rib.20140401.1200.bz2	01-Apr-2014 12:00	5.6M	
 rib.20140401.1400.bz2	01-Apr-2014 14:00	5.6M	
 rib.20140401.1600.bz2	01-Apr-2014 16:00	5.6M	
 rib.20140401.1800.bz2	01-Apr-2014 18:00	5.6M	
 rib.20140401.2000.bz2	01-Apr-2014 20:00	5.6M	
 rib.20140401.2200.bz2	01-Apr-2014 22:00	5.6M	
 rib.20140402.0000.bz2	02-Apr-2014 00:00	5.6M	
 rib.20140402.0200.bz2	02-Apr-2014 02:00	5.6M	
 rib.20140402.0400.bz2	02-Apr-2014 04:00	5.6M	
 rib.20140402.0600.bz2	02-Apr-2014 06:00	5.6M	
 rib.20140402.0800.bz2	02-Apr-2014 08:00	5.6M	
 rib.20140402.1000.bz2	02-Apr-2014 10:00	5.6M	
 rib.20140402.1200.bz2	02-Apr-2014 12:00	5.6M	

- ✦ **MRT = Multi-Threaded Routing Toolkit**
 - ✦ 過去に存在したMRTというルーティングツールで使用されていたデータ形式
 - ✦ MRT自体は廃れたものの、データ形式はZebra/Quaggaで利用されている
 - ✦ しかし、Ciscoなどの商用ルータではサポートされていない...
 - ✦ RFC6396に規定されている
 - ✦ <http://tools.ietf.org/html/rfc6396>

✦ 解説しよう、RFC6396！



ごめんなさい、無理でした・・・orz

✦ quaggaの場合

```
[yasuyuki@yasuyuki ~]$ sudo vtysh

Hello, this is Quagga (version 0.99.15).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

yasuyuki# configure terminal
yasuyuki(config)# dump bgp routes-mrt /var/log/quagga/rib.%Y%m%d.%H%M 60m
yasuyuki(config)# dump bgp updates /var/log/quagga/updates.%Y%m%d.%H%M 15m
yasuyuki(config)# end
yasuyuki #
```

✦ vyattaの場合

```
vyatta@vyatta:~$ sudo vi /etc/quagga/bgpd.conf
dump bgp updates /var/log/quagga/updates.%Y%m%d.%H%M 15m
dump bgp routes-mrt /var/log/quagga/rib.%Y%m%d.%H%M 60m
```



✦ 概要

- ✦ <https://bitbucket.org/ripence/bgpdump/wiki/Home>
- ✦ コマンドはbgpdumpだけど、本当の名前はlibBGPdump？
- ✦ MRTファイル(バイナリデータ)をテキストデータに変換し、分析するためのツールです

✦ インストール

```
$ wget http://www.ris.ripe.net/source/bgpdump/libbgpdump-  
1.4.99.13.tgz  
$ tar xvfz libbgpdump-1.4.99.13.tgz  
$ cd libbgpdump-1.4.99.13  
$ ./configure  
$ make  
$ sudo cp bgpdump /usr/local/bin
```

✦ 使い方の説明

Usage:

```
bgpdump [-m|-M] [-t dump|-t change] [-O <output-file>] <input-file>
```

Output mode:

```
-H      multi-line, human-readable (the default)
-m      one-line per entry with unix timestamps
-M      one-line per entry with human readable
        timestamps
        (there are other differences between -m and -M)
```

Common options:

```
-O <file> output to <file> instead of STDOUT
-s        log to syslog (the default)
-v        log to STDERR
```

✦ 使い方の説明

Options for -m and -M modes:

- t dump timestamps for RIB dumps reflect the time of the dump (the default)
- t change timestamps for RIB dumps reflect the last route modification

Special options:

- T run unit tests and exit

✦ 実際に使ってみる

```
$ bgpdump rib.20140417.0000.bz2
```

```
TIME: 04/17/14 00:00:00  
TYPE: TABLE_DUMP_V2/IPV4_UNICAST  
PREFIX: 1.0.0.0/24  
SEQUENCE: 0  
FROM: 202.249.2.169 AS2497  
ORIGINATED: 01/08/14 23:17:42  
ORIGIN: IGP  
ASPATH: 2497 15169  
NEXT_HOP: 202.249.2.169
```

```
TIME: 04/17/14 00:00:00  
TYPE: TABLE_DUMP_V2/IPV4_UNICAST  
PREFIX: 1.0.0.0/24  
SEQUENCE: 0  
FROM: 202.249.2.86 AS7500  
ORIGINATED: 02/28/14 19:06:57  
ORIGIN: IGP  
ASPATH: 7500 2497 15169  
NEXT_HOP: 202.249.2.86
```

✦ 実際に使ってみる

```
TIME: 04/17/14 00:00:00
TYPE: TABLE_DUMP_V2/IPV4_UNICAST
PREFIX: 1.0.4.0/24
SEQUENCE: 1
FROM: 202.249.2.86 AS7500
ORIGINATED: 03/28/14 05:57:55
ORIGIN: IGP
ASPATH: 7500 2516 6453 7545 56203
NEXT_HOP: 202.249.2.86
```

```
TIME: 04/17/14 00:00:00
TYPE: TABLE_DUMP_V2/IPV4_UNICAST
PREFIX: 1.0.4.0/24
SEQUENCE: 1
FROM: 202.249.2.169 AS2497
ORIGINATED: 03/28/14 05:58:00
ORIGIN: IGP
ASPATH: 2497 6453 7545 56203
NEXT_HOP: 202.249.2.169
```


✦ 実際に使ってみる

```
$ bgpdump -m rib.20140417.0000.bz2
```

```
TABLE_DUMP2|1397692800|B|202.249.2.169|2497|1.0.0.0/24|2497  
15169|IGP|202.249.2.169|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.86|7500|1.0.0.0/24|7500 2497  
15169|IGP|202.249.2.86|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.86|7500|1.0.4.0/24|7500 2516 6453  
7545 56203|IGP|202.249.2.86|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.169|2497|1.0.4.0/24|2497 6453 7545  
56203|IGP|202.249.2.169|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.86|7500|1.0.5.0/24|7500 2516 6453  
7545 56203|IGP|202.249.2.86|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.169|2497|1.0.5.0/24|2497 6453 7545  
56203|IGP|202.249.2.169|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.169|2497|1.0.6.0/24|2497 4826 38803  
56203|IGP|202.249.2.169|0|0||NAG||  
TABLE_DUMP2|1397692800|B|202.249.2.86|7500|1.0.6.0/24|7500 2497 4826  
38803 56203|IGP|202.249.2.86|0|0||NAG||
```

✦ 実際に使ってみる

```
$ bgpdump -M rib.20140417.0000.bz2
```

```
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.169|2497|1.0.0.0/24|2497  
15169|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.86|7500|1.0.0.0/24|7500 2497  
15169|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.86|7500|1.0.4.0/24|7500 2516  
6453 7545 56203|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.169|2497|1.0.4.0/24|2497  
6453 7545 56203|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.86|7500|1.0.5.0/24|7500 2516  
6453 7545 56203|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.169|2497|1.0.5.0/24|2497  
6453 7545 56203|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.169|2497|1.0.6.0/24|2497  
4826 38803 56203|IGP  
TABLE_DUMP_V2|04/17/14 00:00:00|A|202.249.2.86|7500|1.0.6.0/24|7500 2497  
4826 38803 56203|IGP
```

✦ 実際に使ってみる

```
$ bgpdump -M -t change rib.20140417.0000.bz2

TABLE_DUMP_V2|01/08/14 23:17:42|A|202.249.2.169|2497|1.0.0.0/24|2497
15169|IGP
TABLE_DUMP_V2|02/28/14 19:06:57|A|202.249.2.86|7500|1.0.0.0/24|7500 2497
15169|IGP
TABLE_DUMP_V2|03/28/14 05:57:55|A|202.249.2.86|7500|1.0.4.0/24|7500 2516
6453 7545 56203|IGP
TABLE_DUMP_V2|03/28/14 05:58:00|A|202.249.2.169|2497|1.0.4.0/24|2497
6453 7545 56203|IGP
TABLE_DUMP_V2|03/28/14 05:57:55|A|202.249.2.86|7500|1.0.5.0/24|7500 2516
6453 7545 56203|IGP
TABLE_DUMP_V2|03/28/14 05:58:00|A|202.249.2.169|2497|1.0.5.0/24|2497
6453 7545 56203|IGP
TABLE_DUMP_V2|04/03/14 08:40:55|A|202.249.2.169|2497|1.0.6.0/24|2497
4826 38803 56203|IGP
TABLE_DUMP_V2|04/03/14 08:41:20|A|202.249.2.86|7500|1.0.6.0/24|7500 2497
4826 38803 56203|IGP
```

✦ 実際に使ってみる

```
$ bgpdump -M -t change rib.20140417.0000.bz2 |grep ' 18070'

TABLE_DUMP_V2|02/28/14 19:07:09|A|202.249.2.169|2497|117.102.168.0/21|2497 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:07:44|A|202.249.2.86|7500|117.102.168.0/21|7500 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:07:09|A|202.249.2.169|2497|210.158.160.0/20|2497 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:08:11|A|202.249.2.86|7500|210.158.160.0/20|7500 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:07:09|A|202.249.2.169|2497|218.223.32.0/20|2497 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:08:11|A|202.249.2.86|7500|218.223.32.0/20|7500 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:07:09|A|202.249.2.169|2497|221.120.168.0/21|2497 4713
18070|IGP
TABLE_DUMP_V2|02/28/14 19:08:11|A|202.249.2.86|7500|221.120.168.0/21|7500 4713
18070|IGP
TABLE_DUMP_V2|04/16/14 17:45:56|A|2001:200:0:fe00::9d4:0|2516|2400:e000::/32|2516
4713 18070|IGP
TABLE_DUMP_V2|04/16/14 18:21:51|A|2001:200:0:fe00::9c4:11|2500|2400:e000::/32|2500
4713 18070|IGP
```

✦ さきほど、bgpsimpleのところで...

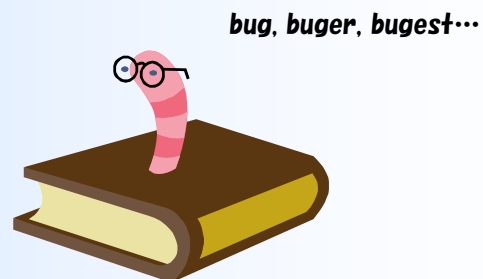
```
$ wget http://archive.routeviews.org/route-views.wide/bgpdata//2014.04/RIBS/rib.20140417.0000.bz2

$ bgpdump -m ./rib.20140417.0000.bz2 > myroutes

$ sudo ./bgp_simple.pl -myas 64512 -myip=192.168.0.1 -peerip=192.168.0.2 -peeras=64513 -holdtime=1800 -keepalive=600 -p myroutes
```

✦ これはつまり、こういうことでした

- ✦ Routeviews ProjectからWIDEルータの経路データを取得
- ✦ one-line per entry with unix timestamps形式で出力
- ✦ 出力したファイルをbgpsimpleの広告経路データとして入力



- ✦ <http://www.bgpview.org/>
 - ✦ 近藤邦昭さんが1998年から開発
 - ✦ 最新版は beta0.42 (2010/07)
- ✦ 対向ルータとpeerを確立し、BGP経路の送受信や各種解析、経路情報のログ出力などができます
 - ✦ 受信経路をIRRで確認したい、送出経路の伝搬時間を計測をしたい
- ✦ 対話型インタフェースによる操作が可能
 - ✦ BGPView Shell (BVS)に接続してコマンドを入力

✦ インストール

```
$ wget http://www.bgpview.org/download/bgpview-beta0.42.tar.gz
$ tar xvfz bgpview-beta0.42.tar.gz
$ cd bgpview-beta0.42
$ ./configure
$ make
$ sudo make install
```


✦ 設定はこんな感じで

```
$ sudo vi /usr/local/etc/bgpview.cfg

LOCAL_AS          64512
IDENTIFIER         192.168.0.1
SYSTEM_LOG        /var/log/bgpview.log
SHELL_PORT        3000
SHELL_MAXCON      4
SHELL_PASSWD      PQAm21Nh3JBF
SHELL_MORE        24
DUMMY_ROUTE_CFG   /usr/local/etc/dummyroute.cfg
NEIGHBOR
    DESCRIPTION    peer1
    ADDRESS        192.168.0.2
    REMOTE_AS      64513
    HOLD_TIMER     600
#    MD5PASS       hogehoge
    4OCTETS_AS     ON
    NLRI           IPv4_UNICAST
```

✦ 設定はこんな感じで

```
VIEW          OPEN
VIEW          UPDATE
VIEW          UPDATTR
VIEW          UPDDATEDUMP
VIEW          NOTIFICATION
VIEW          KEEPALIVE
VIEW          SYSTEM
ANNOUNCE      ROUTE_DIFF
ANNOUNCE      STATUS_CHANGE
LOGNAME        /var/log/bgpview-peer01.log
INTERVAL_INFO 60
INFO           /var/log/bgpview-peer01.info
PREFIXOUT      /var/log/bgpview-peer01.prefix
PREFIXOUTSUM   ON
RECONNECTWAIT 60
```

✦ 起動

```
$ sudo bgpview -d
```

✦ BGPView Shellに接続

```
$ telnet localhost 3000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

BGPView Version 0.42Beta
Copyright (c) 1998-2002 Internet Initiative Japan Inc.
Copyright (c) 2003-2005 Intec NetCore, Inc.
Copyright (c) 2006-2010 Mahoroba Kobo. / Bugest-Network

Password:
BGPView#
```

✦ コマンド一覧(一部抜粋)

```
show users
show proccess
show version
show configuration
show announce dummyroute

show cron list
show ip bgp [detail]
show ip bgp [active|inactive] [detail]
show ip bgp <address>
show ip bgp summary
show ip bgp neighbor <neighbor address>
show ip bgp neighbor <neighbor address> routes
show ip bgp neighbor <neighbor address> [active|inactive]
routes
show ip bgp statics <neighbor address>
```

✦ コマンド一覧(一部抜粋)

```
show ip bgp route time <Time>
show ip bgp route time <Time> le
show ip bgp route time <Time> ge
show ip nexthop summary
show ip nexthop summary neighbor <neighbor address>
show ip route summary

show irr status
show irr status active
show irr status summary
show irr status summary neighbor <neighbor address>

clear ip bgp <neighbor address>
clear ip bgp statics <all | neighbor address>
clear route history
```

✦ コマンド一覧(一部抜粋)

```
do announce dummyroute
do output route filename <Filename>
do output route filename <Filename> neighbor <neighbor address>
do output route filename <Filename> [active|inactive]
do output route filename <Filename> neighbor <neighbor address>
[active|inactive]
do output route filename <Filename> detail
do output route filename <Filename> neighbor <neighbor address>
detail
do output route filename <Filename> neighbor <neighbor address>
[active|inactive] detail

do test route rtt neighbor <Neighbor Address> inject <Injection
Prefix> timeout <Timeout Sec>
do output irrstats filename <File Name> all
do output irrstats filename <File Name> active
do output aslist filename <File Name> neighbor <neighbor
address>
```

✚ 📄 (bgpview-peer01.info)

```
$ cat bgpview-peer01.info
```

Time	Message	Update	Notific	Open	Keepali	Prefix	Withdra	Totalpr
2014/06/06 05:44:24	1	0	0	0	1	0	0	0
2014/06/06 05:45:24	0	0	0	0	0	0	0	0
2014/06/06 05:46:24	0	0	0	0	0	0	0	0
2014/06/06 05:47:24	1	0	0	0	1	0	0	0
2014/06/06 05:48:24	0	0	0	0	0	0	0	0
2014/06/06 05:49:24	0	0	0	0	0	0	0	0
2014/06/06 05:50:24	1	0	0	0	1	0	0	0
2014/06/06 05:51:24	10	10	0	0	0	17	0	0
2014/06/06 05:52:24	0	0	0	0	0	0	0	0

✚ 📄 (bgpview-peer01.log)

```
$ cat bgpview-peer01.log

2014/06/06 05:51:17: BGP Header Type = 2, Length = 54
2014/06/06 05:51:17: GET Message Type = 2 (Update) / Data Length = 54
2014/06/06 05:51:17: DEBUG: Receive Data Size = 35 / Peer = 0
RECV DUMP LEN : 35 Octets
RECV DUMP DATE : 2014/06/06 05:51:17
RECV DUMP 0000 : 0000001B 40010100 5002000C 0205FC42
RECV DUMP 0016 : FC411D4C 09C13B41 400304AC 14014D18
RECV DUMP 0032 : 0100000A
2014/06/06 05:51:17: Receive UPDATE Message
2014/06/06 05:51:17: ===== UPDATE Information =====
Unfeasible Route Length = 0 octets
WITHDRAWN Route is not presented.
Total Path Attribute Length = 27 octets
  Attr. Type: Attr. Flag = 40 / Attr. Type Code = 1
  Attribute : WELL-KNOWN TRANSITIVE COMPLETE
  Attribute Length Normal
  Path Attribute = Origin
  Attribute Length = 1
  ORIGIN Type = IGP
```


✦ ㊦ (bgpview-peer01.log)

```

Attr. Type: Attr. Flag = 50 / Attr. Type Code = 2
Attribute : WELL-KNOWN TRANSITIVE COMPLETE
Attribute Length Enhanced
Path Attribute = AS_PATH
Attribute Length = 12
PATH Segment Type = AS_SEQUENCE : 5
AS_Path = 64513 65001 7500 2497 15169
Attr. Type: Attr. Flag = 40 / Attr. Type Code = 3
Attribute : WELL-KNOWN TRANSITIVE COMPLETE
Attribute Length Normal
Path Attribute = NEXT_HOP
Attribute Length = 4
NEXT_HOP Address = 192.168.0.2
Network Layer Reachability Information Length = 4
Prefix          Next Hop          Metric Loc_Pef Time    AS_Path
1.0.0.0/24      192.168.0.2                    0 64513 65001 7500 2497
15169 i
2014/06/06 05:51:17: ===== UPDATE Information Done =====

```

✦ ㊦ (bgpview-peer01.log)

```
2014/06/06 05:51:17: BGP Header Type = 2, Length = 60
2014/06/06 05:51:17: GET Message Type = 2 (Update) / Data Length = 60
2014/06/06 05:51:17: DEBUG: Receive Data Size = 41 / Peer = 0
RECV DUMP LEN : 41 Octets
RECV DUMP DATE : 2014/06/06 05:51:17
RECV DUMP 0000 : 0000001D 40010100 5002000E 0206FC42
RECV DUMP 0016 : FC4109C1 19351D79 DB8B4003 04AC1401
RECV DUMP 0032 : 4D180100 04180100 05000000
2014/06/06 05:51:17: Receive UPDATE Message
2014/06/06 05:51:17: ===== UPDATE Information =====
Unfeasible Route Length = 0 octets
WITHDRAWN Route is not presented.
Total Path Attribute Length = 29 octets
  Attr. Type: Attr. Flag = 40 / Attr. Type Code = 1
  Attribute : WELL-KNOWN TRANSITIVE COMPLETE
  Attribute Length Normal
  Path Attribute = Origin
  Attribute Length = 1
  ORIGIN Type = IGP
```

✦ ㊦ (bgpview-peer01.log)

```

Attr. Type: Attr. Flag = 50 / Attr. Type Code = 2
Attribute : WELL-KNOWN TRANSITIVE COMPLETE
Attribute Length Enhanced
Path Attribute = AS_PATH
Attribute Length = 14
PATH Segment Type = AS_SEQUENCE : 6
AS_Path = 64513 65001 2497 6453 7545 56203
Attr. Type: Attr. Flag = 40 / Attr. Type Code = 3
Attribute : WELL-KNOWN TRANSITIVE COMPLETE
Attribute Length Normal
Path Attribute = NEXT_HOP
Attribute Length = 4
NEXT_HOP Address = 192.168.0.2
Network Layer Reachability Information Length = 8

```

Prefix	Next Hop	Metric	Loc_Pef	Time	AS_Path
1.0.4.0/24	192.168.0.2	0	64513	65001	2497 6453
7545 56203 i					
1.0.5.0/24	192.168.0.2	0	64513	65001	2497 6453
7545 56203 i					

```

2014/06/06 05:51:17: ===== UPDATE Information Done =====

```

✦ 感想

- ✦ まだあまり試せていないです・・・ごめんなさい
- ✦ なんかうまく動かないところがあるみたい？
 - ✦ IRR関連のコマンドが0.39以降では動かないとか
 - ✦ showコマンドでneighborを指定してもNot foundと言われるとか
 - ✦ CentOSだとbvpasswdがsegmentation faultになるとか
- ✦ フイルート食えない？
 - ✦ kernel: swap_pager: out of swap space
 - ✦ kernel: swap_pager_getswapspace(16): failed
 - ✦ kernel: pid 1413 (bgpview), uid 0, was killed: out of swap space

✦ というわけで...

✦ 近藤さん、そのうちどこかで詳しく教えてください！

$m(_ _)m$

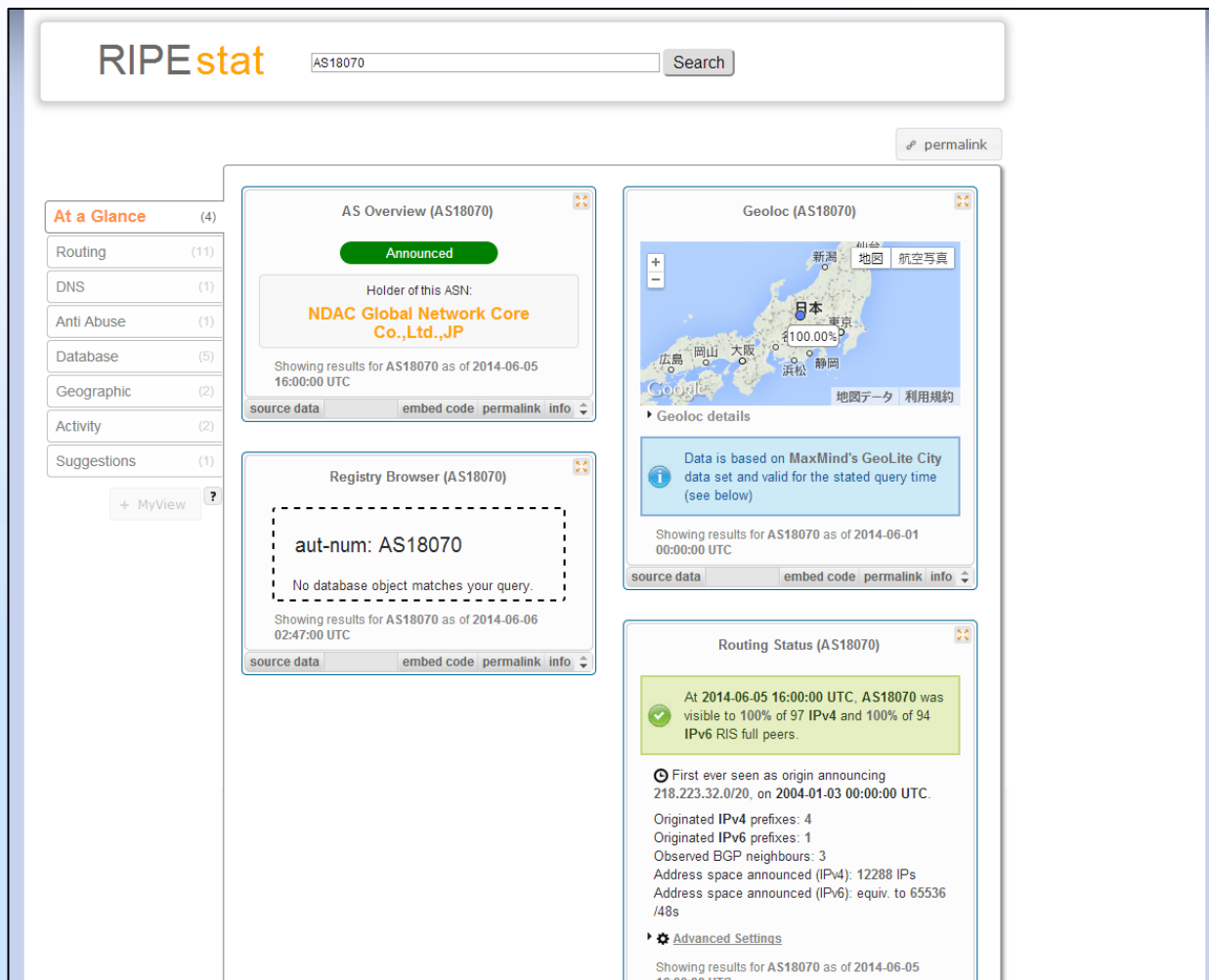


Photo by [Moyan Brenn](#) via Flickr

✦ 概要

- ✦ <https://stat.ripe.net/>
- ✦ RIPE NCCが運営している、番号資源情報や経路広告状況に関するデータ提供サービス
- ✦ ウェブインタフェースで閲覧するほか、APIの提供もあり
- ✦ RIPEstat is a web-based interface that **provides everything you ever wanted to know** about IP address space, Autonomous System Numbers (ASNs), and related information for hostnames and countries **in one place.**
- ✦ (*^▽^*)ステキ！

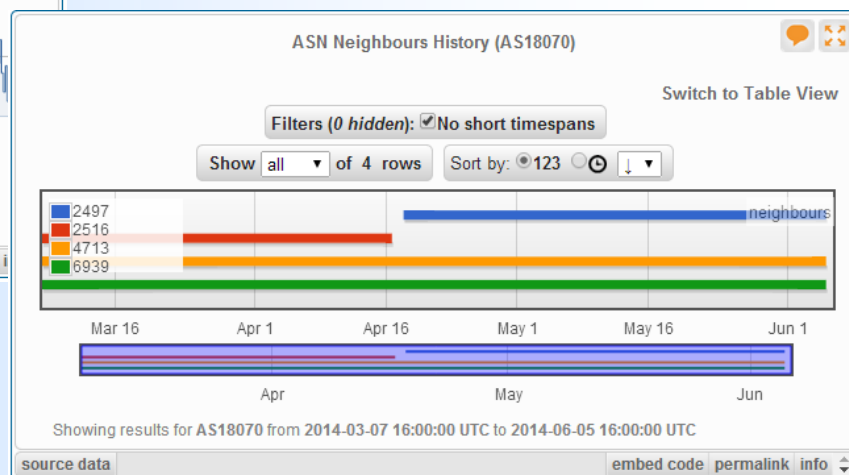
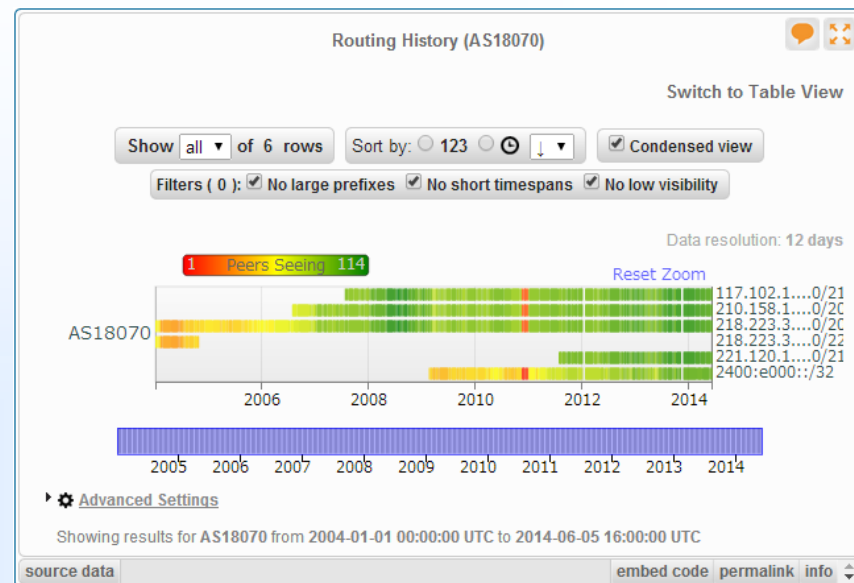
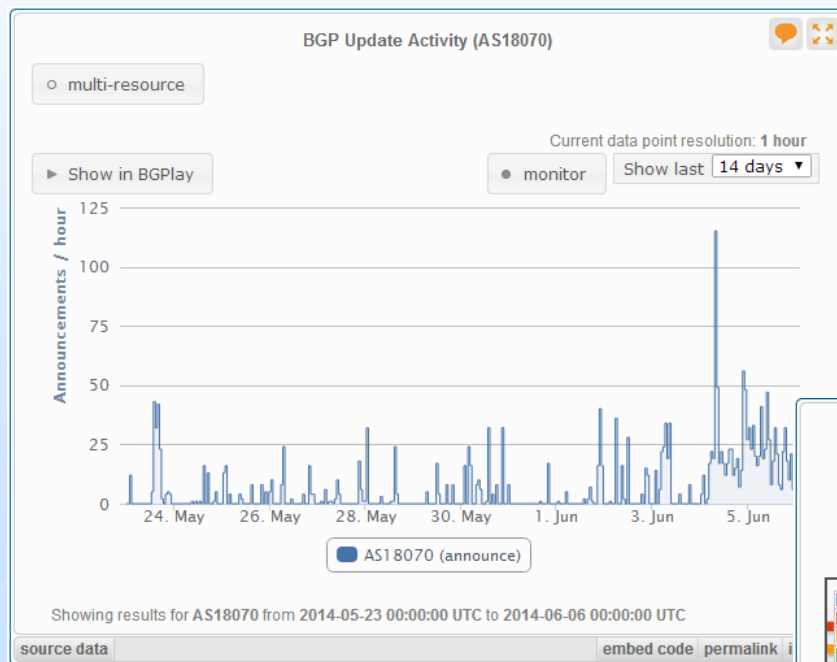
✦ At a Glance



The screenshot displays the RIPEstat interface for AS18070. The top navigation bar includes the RIPEstat logo, a search input field containing 'AS18070', and a 'Search' button. A 'permalink' link is also visible. On the left, a sidebar lists various views: 'At a Glance' (4), 'Routing' (11), 'DNS' (1), 'Anti Abuse' (1), 'Database' (5), 'Geographic' (2), 'Activity' (2), and 'Suggestions' (1). Below these is a '+ MyView ?' button. The main content area is divided into several panels:

- AS Overview (AS18070)**: Features a green 'Announced' button, the text 'Holder of this ASN: NDAC Global Network Core Co.,Ltd.,JP', and a timestamp 'Showing results for AS18070 as of 2014-06-05 16:00:00 UTC'. It includes links for 'source data', 'embed code', 'permalink', and 'info'.
- Registry Browser (AS18070)**: Shows a dashed box with 'aut-num: AS18070' and the message 'No database object matches your query.' It also includes a timestamp 'Showing results for AS18070 as of 2014-06-06 02:47:00 UTC' and links for 'source data', 'embed code', 'permalink', and 'info'.
- Geoloc (AS18070)**: Contains a map of Japan with a location pin, a 'Geoloc details' section, and a note: 'Data is based on MaxMind's GeoLite City data set and valid for the stated query time (see below)'. It includes a timestamp 'Showing results for AS18070 as of 2014-06-01 00:00:00 UTC' and links for 'source data', 'embed code', 'permalink', and 'info'.
- Routing Status (AS18070)**: Displays a green checkmark icon and the text: 'At 2014-06-05 16:00:00 UTC, AS18070 was visible to 100% of 97 IPv4 and 100% of 94 IPv6 RIS full peers.' It lists statistics: 'First ever seen as origin announcing 218.223.32.0/20, on 2004-01-03 00:00:00 UTC.', 'Originated IPv4 prefixes: 4', 'Originated IPv6 prefixes: 1', 'Observed BGP neighbours: 3', 'Address space announced (IPv4): 12288 IPs', and 'Address space announced (IPv6): equiv. to 65536 /48s'. It includes an 'Advanced Settings' link and a timestamp 'Showing results for AS18070 as of 2014-06-05 16:00:00 UTC'.

✦ Routing



Routing

Visibility (AS18070)

AS18070 is visible by 100% of 97 IPv4 and 100% of 94 IPv6 RIS full peers.

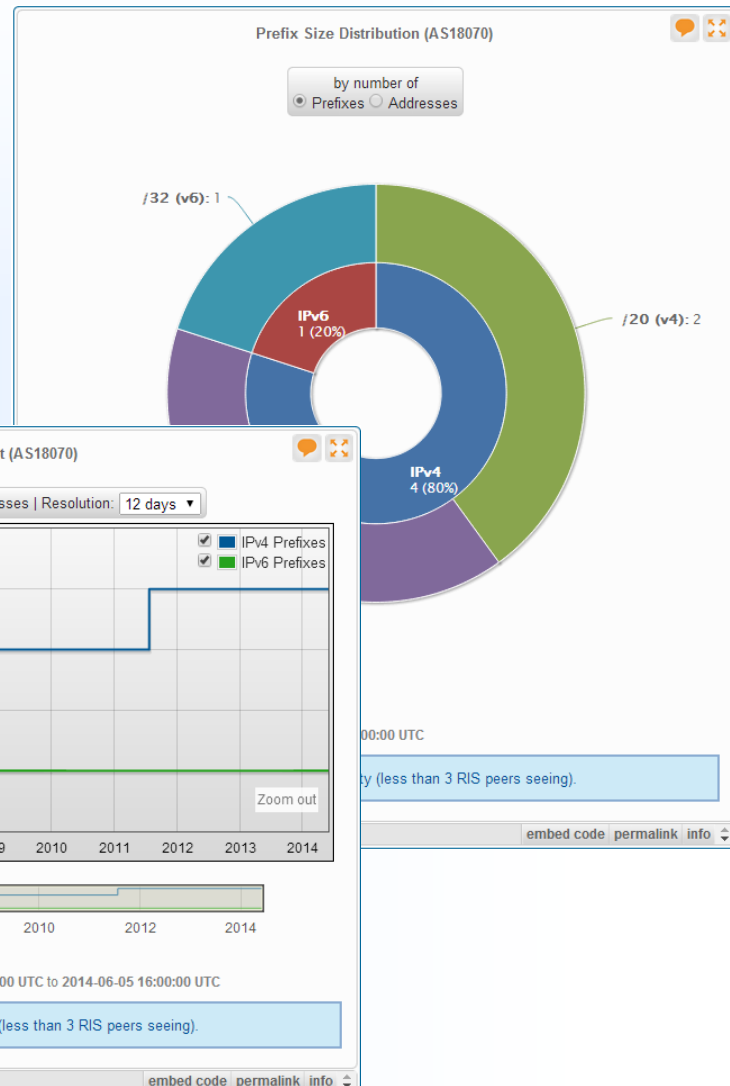
Visibility Location Details of AS18070

RRC	IXP Location	Location	IPv4 peers seeing	IPv6 peers seeing	IPv4 Visibility	IPv6 Visibility
RRC00	RIPE-NCC Multi-hop	Amsterdam, Netherlands	15 of 15	9 of 9	100%	100%
RRC01	LINX	London, United Kingdom	11 of 11	11 of 11	100%	100%
RRC03	AMS-IX / NL-IX / GN-IX	Amsterdam, Netherlands	6 of 6	10 of 10	100%	100%
RRC04	CIXP	Geneva, Switzerland	6 of 6	2 of 2	100%	100%
RRC05	VIX	Vienna, Austria	6 of 6	8 of 8	100%	100%
RRC06	DIX-IE	Tokyo, Japan	1 of 1	1 of 1	100%	100%
RRC07	Netnod	Stockholm, Sweden	2 of 2	4 of 4	100%	100%
RRC10	MIX	Milan, Italy	6 of 6	5 of 5	100%	100%
RRC11	NYIIX	New York City, US	5 of 5	6 of 6	100%	100%
RRC12	DE-CIX	Frankfurt, Germany	16 of 16	23 of 23	100%	100%
RRC13	MSK-IX	Moscow, Russian Federation	10 of 10	4 of 4	100%	100%
RRC14	PAIX	Palo Alto, US	6 of 6	7 of 7	100%	100%
RRC15	PTTMetro	Sao Paulo, Brazil	7 of 7	4 of 4	100%	100%

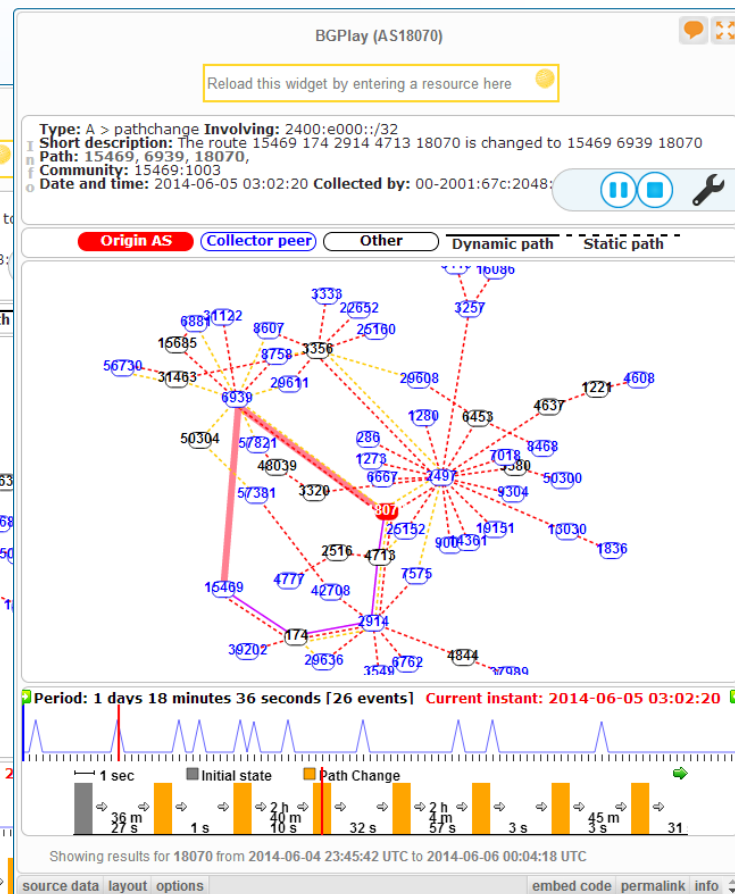
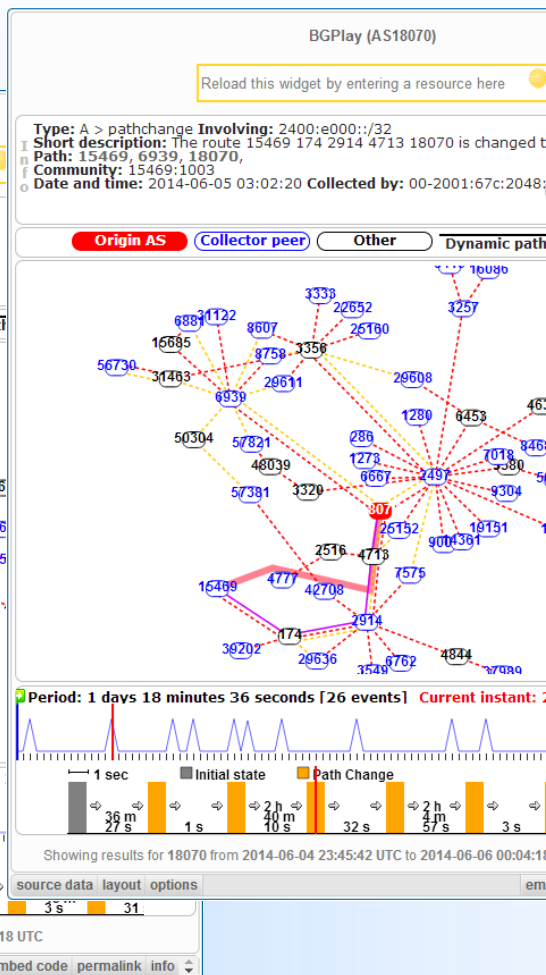
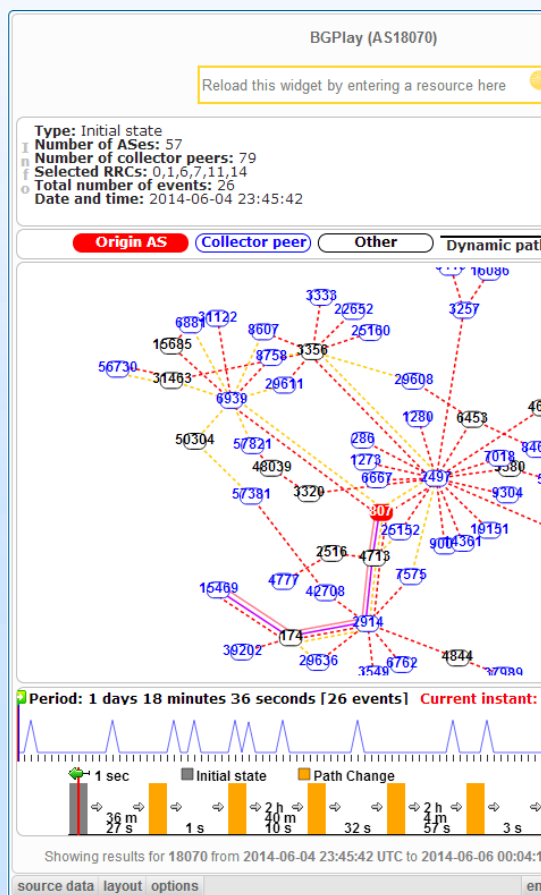
Showing results for AS18070 as of 2014-06-05 16:00:00 UTC

Query time has been set to latest time data is available for (2014-06-05 16:00 UTC)

source data embed code permalink



✦ BGPlay



✦ Use Cases

You are here: [Home](#) > [Data & Tools](#) > [RIPEstat](#) > [Use Cases](#) > [Notable Network Events](#)

Notable Network Events

Analyses of global Internet events using RIPEstat.

Mediterranean Cable Cuts (Jan/Feb 2008)



This report covers the Internet outages caused by the cable cuts in the mediterranean in early 2008.

Youtube - Pakistan Telecom incident (Feb 2008)



This report covers the incident caused by Pakistan Telecom leaking a route from Youtube's AS to the global Internet.

Egyptian Network Outage (Jan 2011)



This report covers the Egyptian Internet Outage that occurred in January 2011.

Libya Internet Outage (Feb 2011)



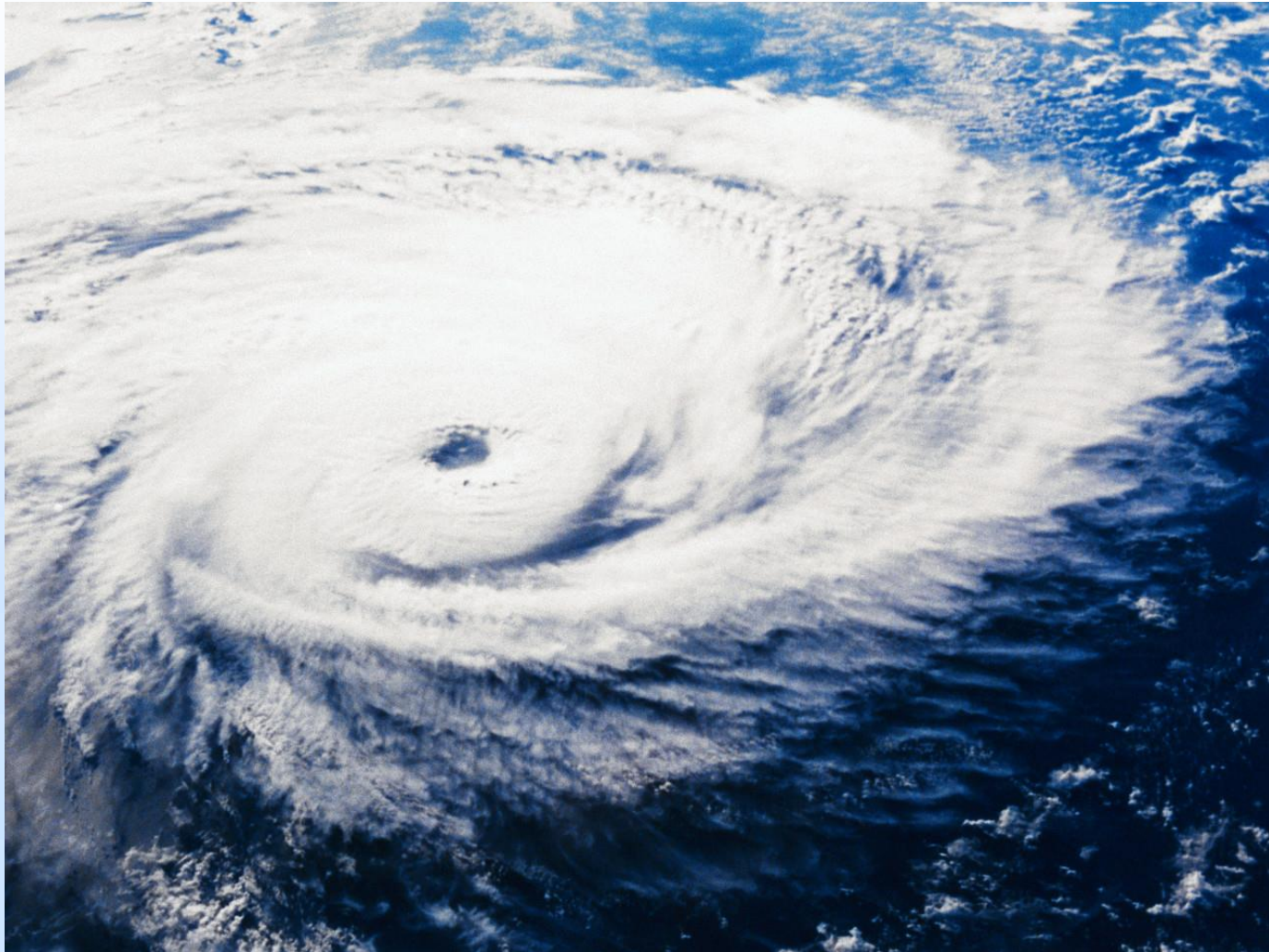
This report covers the Internet Outage in Libya from February 2011.

Japanese Earthquake (Mar 2011)



This report covers the impacts of the 2011 Japanese earthquake on the Internet.

Hurricane Electric BGP toolkit




✦ 概要

- ✦ <http://bgp.he.net/>
- ✦ Hurricane Electricが提供している、経路情報の提供サイト
- ✦ 時間をかけずにパッと調べたいときには便利です

Hurricane Electric BGP toolkit

AS info


**HURRICANE ELECTRIC**
INTERNET SERVICES

AS18070 Global Network Core Co.,Ltd.

Quick Links
[BGP Toolkit Home](#)
[BGP Prefix Report](#)
[BGP Peer Report](#)
[Bogon Routes](#)
[World Report](#)
[Multi Origin Routes](#)
[DNS Report](#)
[Top Host Report](#)
[Internet Statistics](#)
[Looking Glass](#)
[Free IPv6 Tunnel](#)
[IPv6 Certification](#)
[IPv6 Progress](#)
[Going Native](#)
[Contact Us](#)

AS Info | **Graph v4** | **Graph v6** | **Prefixes v4** | **Prefixes v6** | **Peers v4** | **Peers v6** | **Whois** | **IRR**

Company Website: <http://www.global-netcore.jp/>

Country of Origin: [Japan](#) 

Prefixes Originated (all): 5
Prefixes Originated (v4): 4
Prefixes Originated (v6): 1


Prefixes Announced (all): 5
Prefixes Announced (v4): 4
Prefixes Announced (v6): 1

BGP Peers Observed (all): 5
BGP Peers Observed (v4): 5
BGP Peers Observed (v6): 4

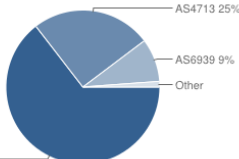
IPs Originated (v4): 12,288
AS Paths Observed (v4): 169
AS Paths Observed (v6): 115

Average AS Path Length (all): 3.778
Average AS Path Length (v4): 3.959
Average AS Path Length (v6): 3.513

Preview
Unavailable

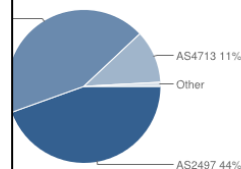
 いいね! 「いいね!」と言っている友達はまだいませ
ふ

AS18070 IPv4 Peers



ASN	Name
AS2497	Internet Initiative Japan Inc.
AS4713	NTT Communications Corporation
AS6939	Hurricane Electric, Inc.

AS18070 IPv6 Peers

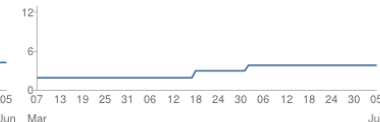


ASN	Name
AS2497	Internet Initiative Japan Inc.
AS6939	Hurricane Electric, Inc.
AS4713	NTT Communications Corporation

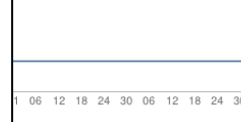
AS18070 IPv4 Peer Count (90 Days)



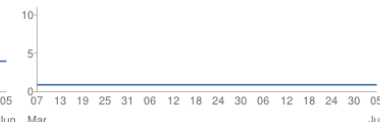
AS18070 IPv6 Peer Count (90 Days)



Pv4 Prefixes Announced (90 Days)



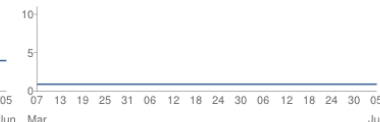
AS18070 IPv6 Prefixes Announced (90 Days)



AS18070 IPv4 Prefixes Originated (90 Days)

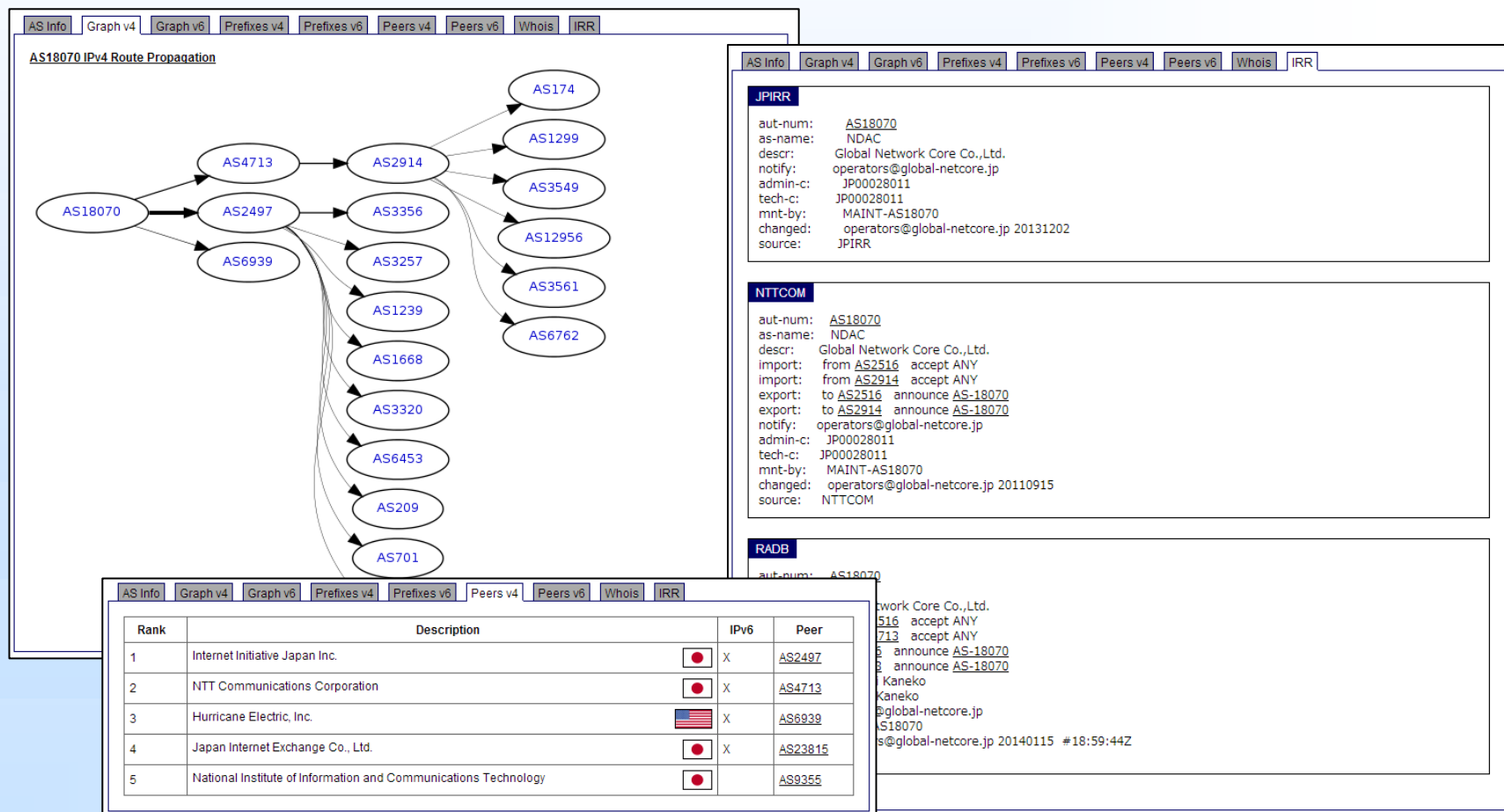


AS18070 IPv6 Prefixes Originated (90 Days)



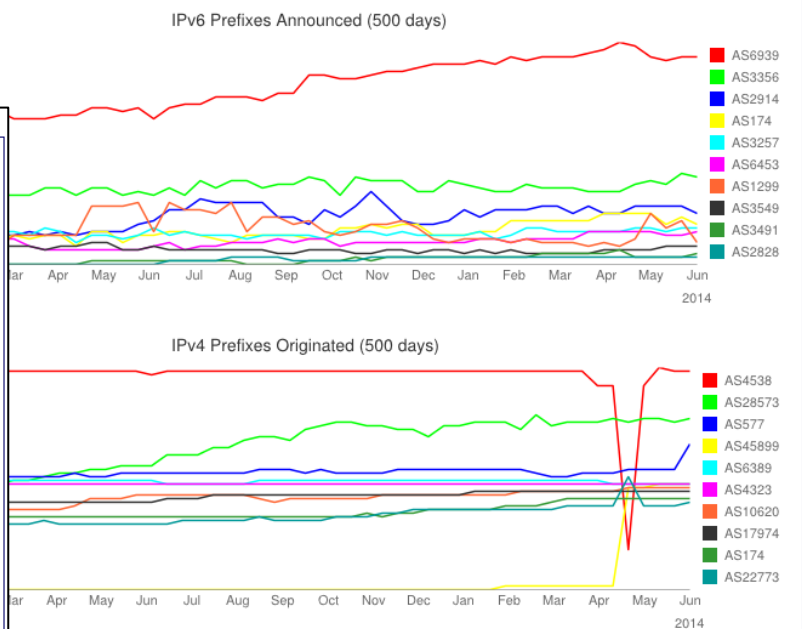
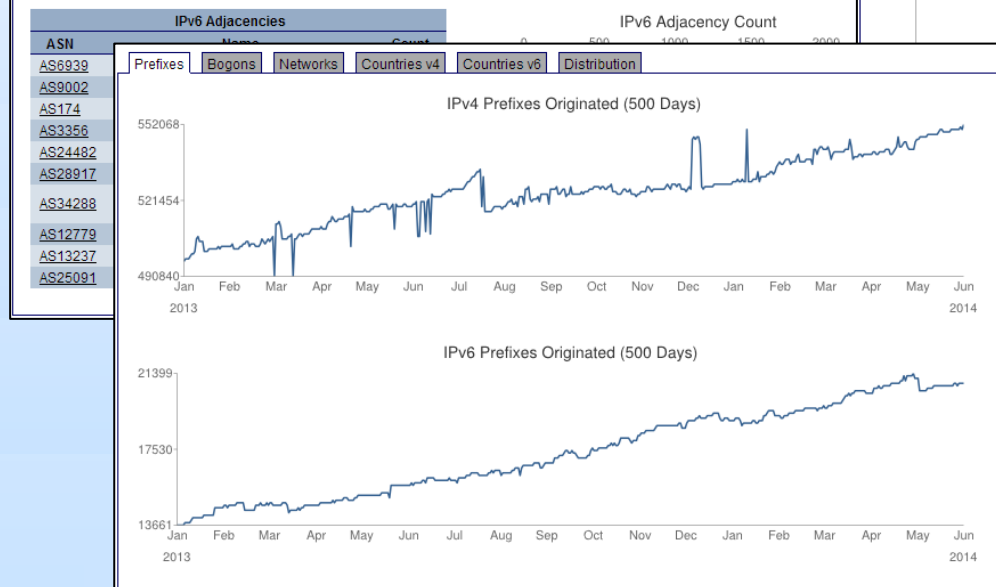
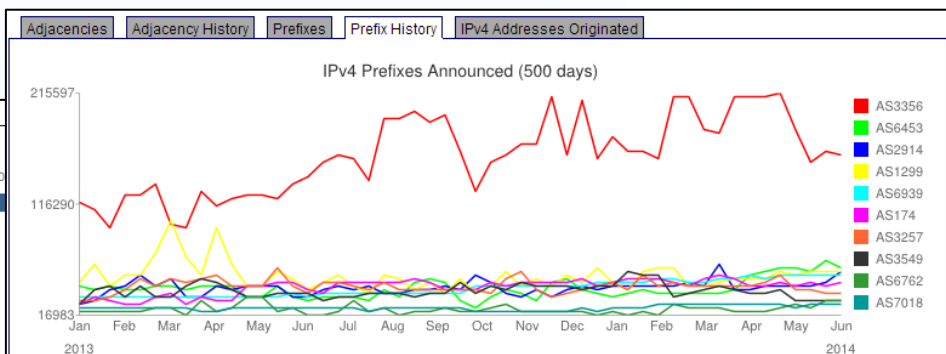
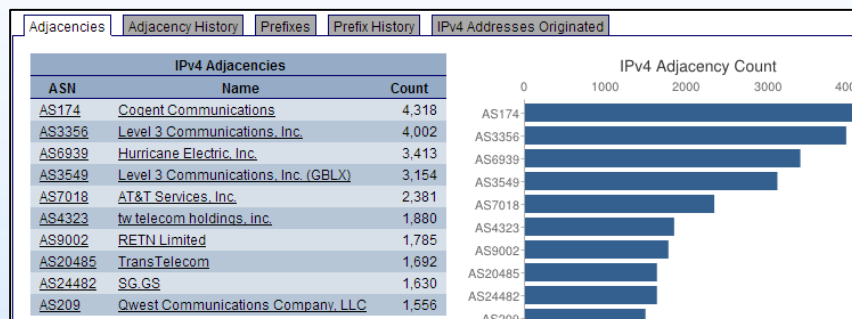
Hurricane Electric BGP toolkit

✦ Graph, Peers, IRR, etc



Hurricane Electric BGP toolkit

一般的な統計情報



- ✦ 以上、BGP関係のツールをいくつかご紹介しました
- ✦ 他にももっといろいろなツールがあるに違いない！
- ✦ いいツールがあったら教えてください♪



✦ でも、本当に大事なものは道具じゃなくて…

✦ 道具を使う

「**目的**」と、「**手法**」と、「**成果**」ですよ！

✦ また明日からも
お仕事がんばりましょう♪



✦ BGP Tools, BGP Software, BGP Utilities

➡ <http://www.bgp4.as/tools>

✦ tools team

➡ <http://tools.bgp4.jp/>

